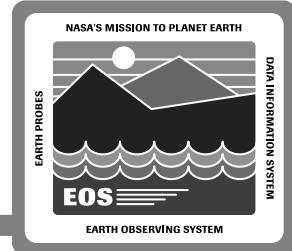


ECS Command Language (ECL)

Rick Broome

17 October 1995

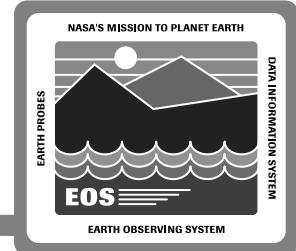
ECL - What is it...



A set of language constructs that allows FOT and IOT members to build directives and procedures

- Directive types:
 - Command
 - Ground
 - Subsystem requests (e.g., display a page)
- Constructs
 - Control (e.g., GO, WAIT, GOTO)
 - Logical structures (e.g., loops, if-then-else, switch)
 - Variables
 - Literals
 - Built-in functions (based upon standard C math library)
 - Comments

ECL Procedures



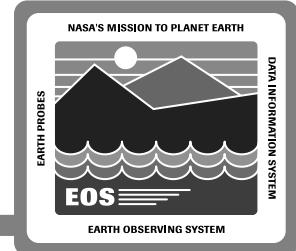
Collection of directives used to perform a specific function

Created using Procedure Builder

Three basic types:

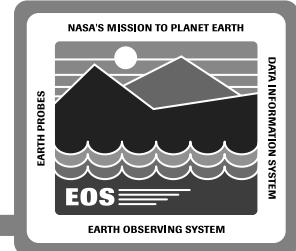
- **Command**
 - Contains at least one spacecraft or instrument command directive
 - Can only be executed by the ground script or the active Command Activity Controller (CAC)
- **Ground**
 - Contains at least one ground configuration directive
 - Can only be executed by the active Ground Controller or CAC
- **Local**
 - Typically performs a function local to the user's workstation
 - Can be executed at any user station or IST

ECL Procedures - Sample



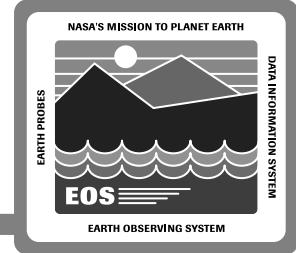
```
PROC Switch_From_CrossTrac_to_Biaxial
#
START:
#
/Set_Quick_Look_Flag      # Turn on the quick look flag
#
IF (@Quick_Look_Flag == 'ON')
{
    /Biaxial_Scan_Mode_Sequence
    #Check APID, data format, BB temp & SWICS w/TLM Verify
    WAIT
    # CAC should select GO when azimuth gimbal & elevation gimbal okay
    WAIT 30    # wait 30 more seconds
    /Unset_Quick_Look_Flag
}
ELSE
{
    PROMPT, "Flag not set - Try Again(requires 3.5 minutes) [Y/N]?", ANS
    IF (ANS == 'Y')
    {
        GOTO START
    }
}
END PROC Switch_From_CrossTrac_to_Biaxial
```

ECL Driving Requirements



- Procedure support**
- Logical constructs (if-then-else, switch, loops)**
- Variables (local, global)**
- Literals - multiple formats**
- Built-in functions**
- Commands (Spacecraft, Instrument)**
- Unix shell commands**
- Comments**

Command Language Study



Evaluated several control center command languages:

- CSTOL PACS STOL
- PSTOL SCL
- TSTOL UIL Specification for Space Station

Evaluated languages against FOS requirements

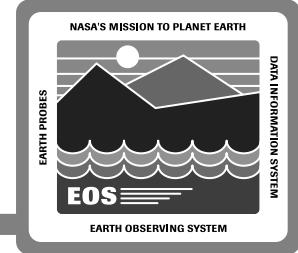
Design goal to be either:

- The I&T language (CSTOL), or
- “Evolvable to” from CSTOL

Took “best” features from evaluated languages

- Examples
 - WAIT directive
 - Loop constructs (While, Until, For)

ECL - Design Features



C-like syntax

Nesting of Procedures

WAIT directive (absolute, relative, conditional)

Parameter references

Directive class (FoEcDirective) shared by FUI and CMS

- Simplifies Ground Script generation process

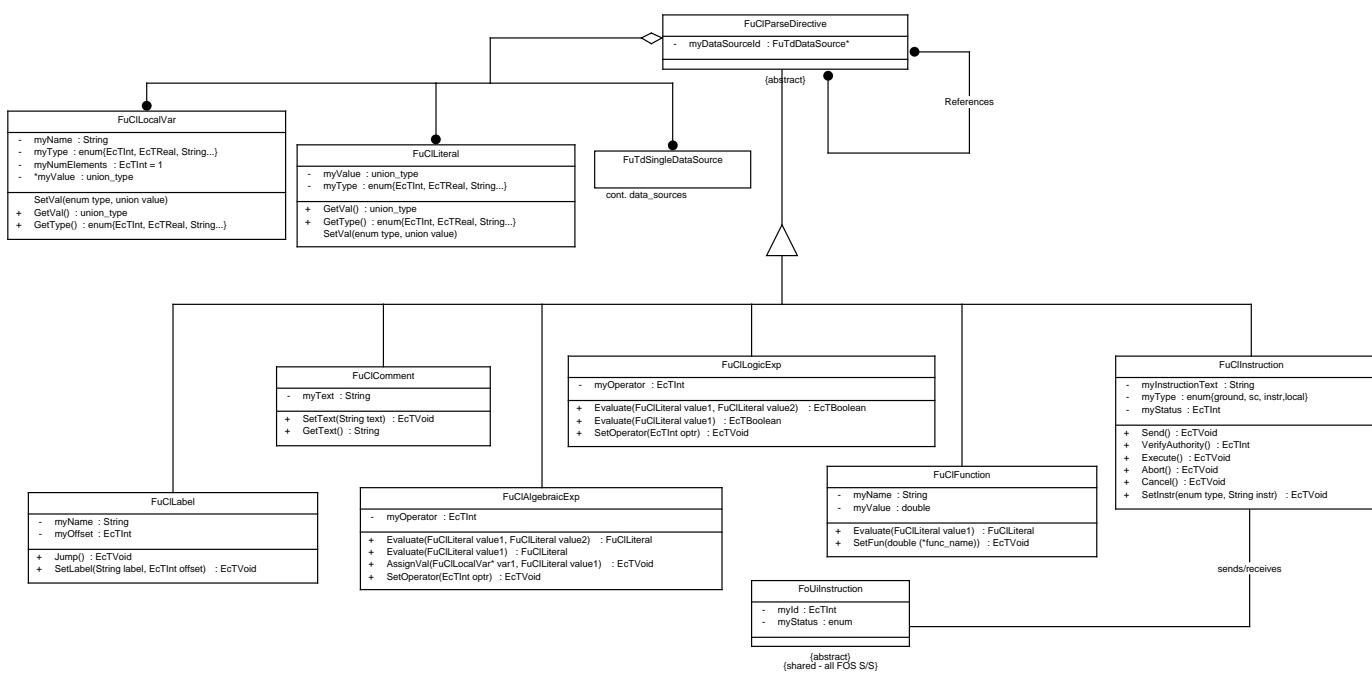
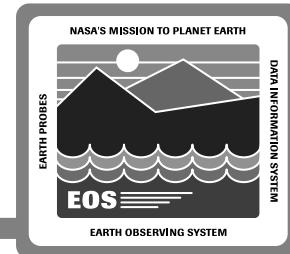
Directive class includes the parser (lex/yacc) operation

Each Directive object understands how to execute itself

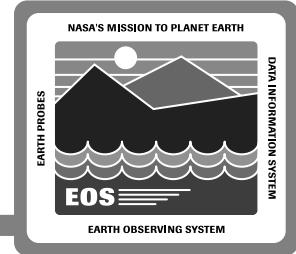
- Simplifies integration of any future directives
- Simplifies testing

Subsystem interfaces (including Command) accomplished via proxies

ECL - Object Model



ECL - Directive Execution



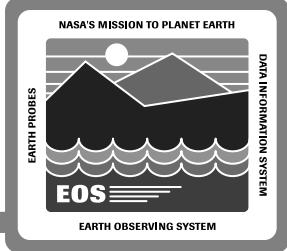
Automated

- From the ground script
- From triggers (local procedures only)

Manual

- Command directives/procedures
 - Command Control window
- Ground/Local directives/procedures
 - Command Control window
 - Control window
 - Procedure Control window

ECL - Directive Handout



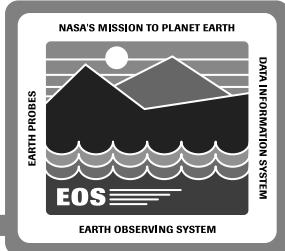
Organized by ECL keyword

Includes

- **Description of each directive**
- **Syntax**
- **Examples**
- **Subsystem that receives the directive**
- **Release in which the directive functionality will be operational**

Reference Book 1, Section 4.4 of the FOS Design Spec (305-CD-040-001) for further information on the ECL structure and syntax

ECL Syntax Example



Keyword	WAIT
Description	Causes execution of a command procedure or ground script to pause. Execution will resume when either the time condition is reached, the expression evaluates to true, or a "GO" directive is encountered. The wait duration may be an absolute time, a relative time, or based upon a conditional statement. It can only be executed from within a procedure.
Syntax	WAIT -- [(<conditionalStatement>) <hh:mm:ss> <sss>]
Example	<p>WAIT (\$x == 1) Waits until the global variable x is equal to 1.</p> <p>WAIT 23:00:00 Waits until the absolute time 23:00:00 is reached.</p> <p>WAIT 30 Waits for 30 seconds.</p> <p>WAIT Waits until a "GO" directive is issued.</p>
Subsystem	FUI
Release	A